IEEE 1588: AN UPDATE ON THE STANDARD AND ITS APPLICATION

John C. Eidson
Agilent Technologies, Inc.
5301 Stevens Creek Boulevard, MS 54U-SM,
Santa Clara, CA 95051-7201, USA
E-mail: *john_eidson@agilent.com*

Abstract

Since the publication of version 1 of IEEE 1588 in November 2002, there has been increasing interest in and development of the technology. The standard has been adopted by a number of trade groups and is referenced in several other standards. The 1588 standard is currently being revised to include provisions for new application areas and to improve the quality of synchronization.

This paper will present the technical issues under discussion in the P1588 standards group and outline the proposed changes to the standard. The applications and technology requirements driving these changes will be discussed. A brief review of application areas, supporting products available, and demonstrated synchronization performance will be provided. The paper will conclude with a brief discussion of how IEEE 1588 is changing electronic measurement instrumentation and how this will change the architecture of measurement systems.

I. WHAT IS IEEE 1588?

IEEE 1588 Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems is a protocol, often called PTP, used to synchronize real-time clocks in modules of a networked distributed system. The protocol was designed specifically to aid in the coordination of activities and the correlation of data in distributed systems typically found in test and measurement, industrial automation, and similar environments. Implementations are easily capable of synchronization to the 10 s of nanosecond range. Sub-nanosecond performance is achievable. Implementations are relatively simple and low cost in both processing and network bandwidth.

II. HISTORY OF THE STANDARD

The protocol is based on work begun at Hewlett Packard and continued at Agilent Technologies. The first version of the standard was published in November 2002 and was described at PTTI in December of that year [1]. In May of 2004, the standard was adopted by the IEC as IEC 61588.

	Form Approved OMB No. 0704-0188
Report Documentation Page	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **01 JAN 2007**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**	
4. TITLE AND SUBTITLE **IEEE 1588: An Update On The Standard And Its Application**		5a. CONTRACT NUMBER	
		5b. GRANT NUMBER	
		5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)		5d. PROJECT NUMBER	
		5e. TASK NUMBER	
		5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Agilent Technologies, Inc. 5301 Stevens Creek Boulevard, MS 54U-SM, Santa Clara, CA 95051-7201, USA**		8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)	
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release, distribution unlimited**			
13. SUPPLEMENTARY NOTES **See also ADM002029.**			
14. ABSTRACT			
15. SUBJECT TERMS			

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT **UU**	18. NUMBER OF PAGES **19**	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**			

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

There have been four international symposia on the standard and its application, the most recent being held at NIST in Gaithersburg in October 2006. The papers presented at these symposia have discussed products and existing and potential applications from several industries, including industrial automation and motion control, test and measurement, military, power generation and distribution, home entertainment, semiconductor manufacturing, and telecommunications. The standard has been tested at plug-fests at each of these symposia and in several other venues. These plug-fests have demonstrated that independent implementations of IEEE 1588 do interoperate and deliver the expected synchronization performance.

There are numerous products available that incorporate IEEE 1588. The largest installation to date is in a new series of turbine controls made by General Electric and currently deployed in several power plants around the world. The General Electric design and prototypes were presented at the symposium in 2003 **[2]**. The current products on the market that incorporate IEEE 1588 include microprocessors from six manufacturers, GPS-linked master clocks from three vendors, boundary clocks (1588-enabled Ethernet switches), NIC cards, software protocol stacks, RF downconverters, and aircraft flight monitoring instrumentation.

IEEE 1588 is included or referenced in several standards, including IEEE 802.1as, PC37.1, IEEE 1646-2004, and those of the LXI Consortium─ *http://www.lxistandard.org/home*, and the ODVA─ *http://www.odva.org/*.

Additional information on the standard, future symposia, products, and pointers to recent literature may be found on the IEEE 1588 Web site hosted by NIST at *http://ieee1588.nist.gov*.

III. IEEE 1588 BASICS

Synchronization using IEEE 1588 is a two step process: establishing a master-slave hierarchy, and synchronization.

The master-slave hierarchy is established by an exchange of multicast messages between devices. Each device runs an independent copy of the protocol's best master clock algorithm, or BMC. The messages, which in version 1 of the standard are called Sync messages, contain network path length data and information describing the quality of the clock considered the "best clock in the system," or grandmaster, by the sending device. In each device, the BMC compares this information with similar information received via other network ports and with its own characteristics to determine the state of each of the device's ports. In simple systems, this state will be either master or slave. The device will synchronize to the clock seen on the slave port and will serve as a master to devices downstream of a master port. If the device considers itself to be better than any clock described in the received Sync messages, it will place all of its ports in the master state and will be the grandmaster clock for the system.

For each master-slave pair, the slave port will synchronize its device's clock to that of the master port. This is accomplished by a two-way exchange of timing messages, as illustrated in Figure 1. The timing messages, Sync and Delay_Req, are timestamped at transmission and reception. The timestamps generated by the master device of the pair are communicated to the slave in the Follow_Up and Delay_Resp messages respectively. The slave uses the four timestamps to compute the mean path delay and the offset between the two clocks as follows:

$$offset\_from\_master = (t_2 - t_1) - mean\_path\_delay$$

$$mean\_path\_delay = [(t_2 - t_1) + (t_4 - t_3)]/2$$

These messages are exchanged roughly once a second in current Ethernet implementations. Although the protocol is designed to run on any packet-based network, all implementations to date have been on 10/100 or 1000BT Ethernet. The remainder of this paper discusses only Ethernet implementation issues.

The primary causes of synchronization degradation are the resolution of the clocks, the characteristics of the local oscillator, latency bias and fluctuations in the protocol stacks, and network latency bias and fluctuations. Most 1588 devices reduce protocol stack fluctuation effects by generating the timestamps at the MII/GMII interface between the PHY and MAC, as illustrated in Figure 2. The residual PHY fluctuations are not significant in synchronizing to 10 s of nanoseconds, but will be significant at higher accuracy. Network components such as switches and routers introduce fluctuations ranging from 100 s of microseconds to milliseconds or more due to message queuing. The boundary clock specified in IEEE 1588 and illustrated in Figure 2 removes these fluctuations by eliminating the queues for the timing messages. All PTP timing-related messages terminate at special circuitry at the ports of boundary clocks. Master or slave clocks connected to the switch synchronize to the clock in the switch as illustrated. The synchronization performance achievable using boundary clocks has been shown to be independent of network traffic to at least 90% loading on the network, a considerable advantage in designing systems. Ethernet boundary clocks are available and more vendors are expected to enter the market.

Synchronization between a grandmaster clock and a slave communicating via a boundary clock as illustrated in Figure 2 results in an approximately normal distribution with no outliers outside of ± 240 ns over an 84-hour period in a system carrying normal Ethernet traffic found in commercial installations [3]. If the boundary clock is the grandmaster, connected end devices will be within ± 120 ns of the grandmaster [3]. The performance in these measurements was limited by the resolution of the clocks and the quality of the oscillators.

A hint of future performance is illustrated in Figure 3. This figure shows the synchronization between two clocks communicating over a 100BT Ethernet cross-over cable using a clock design with an effective LSB of 1 ns. The measured standard deviation is 0.771 ns, all outliers within +1.6 to ─2.4 ns and a mean of 0.31 ns [4]. High-quality oscillators were used and the performance is limited by the LSB and possibly residual PHY fluctuations.

IV. IEEE 1588 VERSION 2

Following discussions at the 2004 Conference on IEEE 1588, a PAR was approved by the IEEE for enhancements to the original standard to meet higher performance requirements and the requirements of a much larger set of potential applications than anticipated in the design of version 1.

The main goals outlined in the PAR for the P1588 committee are:

1. Enable sub-nanosecond synchronization. This goal is driven primarily by military and test and measurement applications.
2. Shorten timing messages and make them of equal length. This is a requirement for use in telecommunication where packets of varying length will suffer different latency in traversing network devices.
3. Enable long linear network topologies. The cascading of boundary clocks, in effect cascading the servo loops in each, will degrade performance in chains of up to 100 devices typical of

installations in industrial automation.
4. Provide certain fault tolerance features. The principal requirement is for rapid recovery from network reconfiguration and for the failure of a grandmaster clock. This is particularly important in industrial control environments and other critical applications.
5. Provide mappings of the protocol for several non-Ethernet networks, for direct layer 2 Ethernet, and for IPv6.

Version 2 is expected to be ready for ballot by mid-February 2007.

The remainder of this section describes the changes in IEEE 1588 to meet these goals.

CHANGES IN THE PACKET FORMATS AND MESSAGE TYPES

In version 1, the Sync packet contains both timing information and the data needed for the operation of the best master clock algorithm. In order to shorten this message, these two functions have been separated into two version 2 messages: Announce and Sync. This allows all timing messages to be designed with the same length.

In addition, the split permits different message rates for timing messages used in synchronization and for the Announce messages used for establishing the master-slave hierarchy. This feature is essential for applications that require high accuracy or for those that employ very inexpensive crystal oscillators as it permits timing sampling intervals matched to the wander characteristics of the crystals and simplifies the design of the servos in slave clocks.

The version 2 timing messages are structured as illustrated in Table 1.

Table 1. Sync and Delay_Req timing message format.

Bits								Octets	Offset
7	6	5	4	3	2	1	0		
transportSpecific				messageId				1	0
Reserved				versionPTP				1	1
messageLength								2	2
domainNumber								1	4
reserved								1	5
flags								2	6
correctionField								8	8
reserved								4	16
sourcePortIdentity								10	20
sequenceId								2	30
control								1	32
logMessagePeriod								1	33
originTimestamp								12	34

The first 34 octets comprise a header common to all version 2 PTP messages.

This header is designed to enable all known hardware designed to version 1 to be used in at least the IPv4

mapping of version 2. Version 1 software will need to be upgraded to reflect changes in the format and in some cases data types. In addition, there has been some simplification of the best master clock algorithm and message processing.

The two key timing fields in the Sync and Delay_Req timing messages are the correctionField and the originTimestamp. The timestamp data type has a resolution of only 1 ns as in version 1. To accommodate higher timing precision, the correctionField data type has a resolution of 2^{-16} ns. Any fractional nanosecond portion of a timestamp is recorded in this correctionField by the sending device. A port receiving a timing measurement computes the actual timestamp by summing the originTimestamp and correctionField.

The version 2 Announce message contains the common header plus the information required by the BMC that was carried in the version 1 Sync messages. The remaining version 1 messages Delay_Resp and Management have also been modified in version 2 as a result of the changes introduced in the common header. The processing of management messages has also been simplified to a simple get, set, acknowledge semantics. This allows many of the version 1 management messages to be combined with the different functions embodied in the data carried.

In addition to the new Announce message, version 2 introduces a set of messages for the peer delay mechanism described later, signaling messages for direct clock-to-clock messaging, and a formal mechanism for extending any message based on a type, length, value (TLV) semantics.

END-TO-END TRANSPARENT CLOCKS

To enable long linear topologies without the penalty of cascaded servo loops, version 2 specifies two forms of transparent clocks, end-to-end and peer-to-peer. A block diagram of an end-to-end transparent clock is shown in Figure 4.

In a boundary clock, all PTP timing messages terminate at the input port and synchronization occurs between the boundary clock and the attached device. In an end-to-end transparent clock, all PTP messages pass through the switch according to the addressing rules of the network. However, all PTP timing messages are modified to correct for the time spent traversing the switch, the residence time.

As shown in Figure 4, a timestamp for each timing message is generated at the ingress port based on a local clock in the end-to-end device. An additional timestamp is generated at the egress port as the timing message leaves the device. The residence time is the difference between these two timestamps. To ensure that the residence time is accurate, the local clock in the end-to-end device is syntonized, i.e. shares the same definition of the second, to the grandmaster clock in the system. Syntonization requires much less effort than the synchronization required in a boundary clock. Syntonization is required to maintain high accuracy. For example, if the timing message takes 1 ms to traverse the switch and the oscillator specifications are 100 ppm, the resulting error in the computed residence time could be as much as 100 ns, $(10^{-3} \times 10^{-4} \text{s})$.

The residence time correction is inserted into the correctionField of an outgoing message, as illustrated in Figure 5. Some devices, called one-step clocks in version 2, implement this entire process in hardware, which enables the correction to be inserted in the timing message, e.g. a Sync message, on the fly. An alternate design, the two-step clock, allows the residence time correction to be inserted into the correctionField of a following message, for example the Follow_Up message for a Sync message.

The device receiving a timing message computes the effective timestamp by summing the originTimestamp field and the correctionFields of the timing message and any associated follow up messages. The net result is that to the receiver, the residence times have been removed and the network appears as though the end-to-end clocks were not present, hence the term transparent clock.

End-to-end transparent clocks can be cascaded without the penalty of cascaded servo loops. They are also much simpler to design than a boundary clock, since they do not have to synchronize the local clock and more importantly maintain all of the port and internal state mechanisms required by the BMC. The penalty is that the master clock will see all slaves connected to the end-to-end devices, which increases the processing load on the master.

In a system containing end-to-end transparent clocks, synchronization between a master and slave uses the Sync, Delay_Req, Follow_Up and Delay_Resp messages in exactly the same way as depicted in Figure 1. The only difference is that the residence times in the end-to-end clocks will not appear in the values computed for mean_path_delay or in the $(t_2 - t_1)$ portion of the offset_from_master, as discussed above.

End-to-end transparent clocks have been prototyped in 1000BT Ethernet and have been shown to operate in plug-fest testing.

PEER-TO-PEER TRANSPARENT CLOCKS

The peer-to-peer clock works exactly like the end-to-end transparent clock, except that in addition to correcting timing messages for residence time, the peer-to-peer clock also corrects for the network link latency on the ingress port.

The link latency is measured using the peer-to-peer delay mechanism defined in version 2. Three new messages, Pdelay_Req, Pdelay_Resp, and Pdelay_Resp_Follow_Up, are used. Functionally, these messages are analogous to and perform exactly like the Delay_Req, Sync and Follow_Up messages, as shown in Figure 6. The mean path delay is computed as:

$$mean\_path\_delay = [(t_2 - t_1) + (t_4 - t_3)] / 2$$

The details of which fields of the peer messages contain the timestamp information from the responder depend on whether the clocks are one-step or two-step clocks. The two responder timestamps will be returned either as two separate timestamps or as the difference. As in the case of the end-to-end clock, syntonization of the local clock is needed to ensure high system timing accuracy.

A block diagram of a peer-to-peer transparent clock is shown in Figure 7. In addition to the functions in the end-to-end clock, there is the additional per port "Pdelay Req/Resp mechanism" block responsible for the exchange of the peer messages with an identical block in the port at the other end of the network link connecting requestors and responders. Both ends of a link independently execute this peer-to-peer measurement. Thus, the ports on both ends of the link will know the link latency. Furthermore, this link measurement is executed on all ports of the device including those that may be blocked by an underlying spanning tree protocol.

Because peer-to-peer clocks measure network path delay using the peer messages, a peer-to-peer clock does not forward Delay_Req or Delay_Resp messages. Thus, peer-to-peer clocks must be used in a homogeneous system in which all clocks implement the peer mechanism. The peer-to-peer clock corrects

Sync messages for both residence time and link delay, as illustrated in Figure 8. These corrections are inserted into the correctionField of either the Sync or Follow_Up messages, depending on whether the clocks are one-step or two-step clocks.

Peer-to-peer clocks are clearly more complex to implement than an end-to-end clock. They eliminate the need for the Delay_Req and Delay_Resp messages used to compute mean path delay. Therefore in a peer-to-peer system the master clock does not receive any timing messages from slaves, which reduces the processing load on the master.

However, the principal advantage of the peer-to-peer transparent clock is its ability to recover rapidly from changes in network connectivity. If the network topology changes and Sync messages appear on a different port, the needed timing corrections can be made using the previously measured link delay on that port. In contrast, a system with end-to-end clock must wait for a new path delay to be computed using the delay response messages. In many industrial automation systems, the ability to reconfigure rapidly and maintain timing accuracy is critical.

ADDITIONAL VERSION 2 FEATURES

There are several other new features that are expected to be included in version 2 of IEEE 1588.

In version 1, all PTP messaging was conducted via multicast transmission. In version 2, a mechanism is provided for a slave clock to negotiate unicast transmissions with the master. These unicast transmissions will be permitted during a fixed time interval and can be more frequent than permitted for multicast transmissions. This feature is expected to be widely used in telecommunications applications.

Mappings to three industrial protocols are likely: DeviceNet™, ControlNet™, and PROFINET. These protocols are widely used in industrial automation. DeviceNet and ControlNet use a CAN-based physical layer, while PROFINET uses an Ethernet physical layer. In addition, a mapping to layer 2 Ethernet will be specified. This mapping has been assigned a new Ethertype by the IEEE. It is expected to see application in industrial automation, the power industry, and in telecommunications. Finally, for applications that currently use the version 1 mapping to UDP/IPv4, a mapping to IPv6 is specified.

In some applications, for example telecommunications, system designers want to explicitly control the master-slave hierarchy design. The "preferred" flag of version 1 has been expanded into a field in version 2 which will allow designers to explicitly order the selection of the grandmaster clock. There will also be provision for configuring acceptable master clocks and for monitoring clocks that are not the current grandmaster clock in the system. This latter provision enables suitably designed slaves to predetermine the timing differences between the current grandmaster and alternates in the event of a failure or network reconfiguration. There will also be a mechanism for designating a cluster of potential grandmaster clocks and a mechanism for rapid changeover in the event of a failure.

There are several other topics being discussed in the P1588 committee, but it is far from certain that solutions will be found in time for inclusion in version 2. The major unresolved topic is that of security. Two security issues are proving difficult. The first is exactly what requirements must be met. With the very diverse set of interests in IEEE 1588 comes an equally diverse set of security requirements, including some that do not want any burden on devices to support security. The second difficulty is that all mechanisms considered are difficult to implement while maintaining the accurate timing performance which is the hallmark of IEEE 1588.

V. APPLICATIONS

The interest in IEEE 1588 is quite broad, as evidenced by the diverse membership of the P1588 working group, the participation in the IEEE 1588 symposia, the literature, and known products in development.

INDUSTRIAL AUTOMATION AND MOTION CONTROL

This industry was the first to adopt IEEE 1588 and has continued to support and advance the technology. It is almost certain that this group will continue to use the technology. The driving force is motion control where coordinated actions are required. IEEE 1588 is typically used to establish a time-slotted (TDMA) communication model in which control updates are sent at regular intervals. In some situations, timestamping of data is required for control, operation monitoring, and, occasionally, legal reasons. As noted, General Electric already has IEEE 1588 enabled products in the field. Two major vendors in the industry, Siemens and Rockwell Automation, have demonstrated prototypes of IEEE 1588 products at every IEEE 1588 symposium, including version 2 devices at the most recent plug-fest.

The transparent clocks, both end-to-end and peer-to-peer, are expected to find wide use in this industry.

HOME AUDIO-VISUAL ENTERTAINMENT

The IEEE 802.1as committee is preparing a standard for devices to distribute high quality audio and visual data in a home Ethernet environment. They have selected version 2 clocks as the basis for these devices. The accuracy requirements driving 802.1as are absolute timing to better than a microsecond and jitter less than 100 ns filterable to 100 ps **[5]**. These specifications appear to be well within the capability of IEEE 1588. Since cost is a major consideration, IEEE 802.1as will probably specify a sampling rate somewhat higher than other areas to permit the use of cheaper oscillators.

The IEEE 802.1as standard has the backing of several large and many smaller companies. Since this is a consumer products application, it is reasonable to expect that low-cost silicon implementing IEEE 1588 version 2 clocks devices will be forthcoming.

TEST AND MEASUREMENT

Traditional test and measurement systems have been based on instrumentation using IEEE 488, more commonly known as GPIB. These systems invariably use a remote procedure call model of communication between a central controller and one or more instruments. Today, many new instruments are designed with an Ethernet interface. Based on this trend, the LXI Consortium has created a standard specifying how instruments interact over a LAN; see *http://www.lxistandard.org/home/*.

The LXI Class B devices specify the use of IEEE 1588 to create a system-wide precise timescale for use in timestamping data and coordinating system activities. In addition to IEEE 1588, Class B devices provide for peer-to-peer messaging, LAN and time-based trigger models, and the ability to include application-specific code and configuration in each instrument. Taken together, these features permit a distributed model of program execution unavailable in current instrument systems **[6]**.

Figure 9 illustrates an LXI Class B-based test and measurement system. Each of the devices contains an application script executed based on timestamps distributed in peer-to-peer event messages. The specified times are compared to the local IEEE 1588 clock to determine the exact time of execution.

Timing accuracies of a nanosecond or below will enable accurate coordination of system events, triggers, and data correlation that today require careful attention to the lengths of trigger cables and hand tuning of controller programs **[6]**.

To date, there are several announced Class B instrumentation products, mostly high-end RF measuring equipment. It is far from clear how Class B instrumentation will evolve and what kinds of systems will make use of the technology. However, IEEE 1588 is a clear benefit for high-count data acquisition measurement applications, as it enables the timestamping of data at the point of acquisition without the expense of distributing proprietary timing circuits or the use of protocols like IRIG-B as is common today.

MILITARY AND AEROSPACE

The military and aerospace industries have been investigating IEEE 1588 technology since its introduction in 2002 **[7-9]**. At the plug-fest at the 2006 IEEE 1588 conference, two vendors demonstrated stackable measurement modules designed for onboard aircraft monitoring during flight qualification. These devices incorporate IEEE 1588 to timestamp data for post-acquisition analysis and correlation with flight data.

Clearly, high-accuracy timing is required for many military test and operational applications. Nanosecond timing is no doubt sufficient for mechanical or sound-based systems, but for RF devices, test ranges, high- speed aircraft and missiles and similar areas, sub-nanosecond timing is probably a realistic requirement. The days where the military-funded devices designed specifically to meet military applications are probably over. The IEEE 1588 adoption rate in military applications probably depends on how rapidly commercial silicon and devices appear on the market that can support military timing needs.

POWER DISTRIBUTION

The driving application in this field is the roll-out of wide-area synchrophasor measurement systems in the power grid. These systems measure the local phase on the grid against an absolute time reference. By comparing data collected across the grid, the operators are able to provide better control of distribution parameters. Absolute accuracy to roughly a microsecond is required for this application **[10]**. One proposed implementation is to use IEEE 1588 running on the LAN in a sub-station to replace IRIG-B for distributing time **[10]**. Sub-stations across the grid would then synchronize using GPS. The deployment of IEEE 1588 in this application depends on the availability of the necessary IEEE 1588 Boundary or Transparent Clocks in a rugged form factor.

TELECOMMUNICATIONS

There have been many different ideas discussed about how IEEE 1588 will be used in telecommunications **[11]**. There is no question that the driving force is the shift from circuit-based to packet-based communications and the rise of streaming video and similar high-data-rate time-sensitive applications. There are ongoing large scale field trials of IEEE 1588 technology in this area **[12]**. The synchronization of node-B sites in cellular communication systems is another often-mentioned application. Yet of all the application areas, it remains the most problematic, although industry participation in the IEEE 1588 standardization effort has been massive.

Perhaps the most interesting IEEE-1588-related discussions in the telecommunication industry are taking place in the ITU **[13]**. These discussions deal with the problem of recovering timing in core networks based on Ethernet that are expected to replace SONET cores in many situations. This is an extremely controversial area **[14]**.

One proposal is to make the physical layer of Ethernet synchronous throughout the network by configuring switches into a master-slave hierarchy. Each node in this hierarchy then frequency-locks to the Ethernet continuous symbol stream transmitted by its master, as illustrated in Figure 10. At each node, the recovered Tx clock is used to synchronize a precision local oscillator that then drives both the return path and the downstream link. In this way, very precise frequency transfer is achieved. The proposal then calls for IEEE 1588 to be overlaid on top of this "synchronous" Ethernet physical layer for purposes of time transfer. It is known that poor transfer of frequency, essentially the elimination of wander, is a primary source of timing error using IEEE 1588. Experiments using a common oscillator in all nodes show that IEEE 1588 time transfer over Ethernet is limited primarily by the resolutions of the clocks. There is good reason to believe that, using the ITU-proposed scheme, IEEE 1588 time transfer accuracy can approach 100 ps **[15]**. If this is indeed true, the effect on test and measurement, military, and telecommunications applications would be enormous.

VI. CONCLUSIONS

IEEE 1588 technology continues to evolve and improve. When version 2 of the standard is complete in early 2007, it is expected that there will be a marked upswing in the number of products announced. There is no doubt that interest in the technology is increasing as it becomes more widely known. While many of the proposed applications will no doubt fall by the wayside, it seems certain that IEEE 1588 will find significant use in industrial automation, data acquisition, and IEEE 802.1as systems. The ultimate usage in test and measurement, power distribution, military, and most of all telecommunications awaits the efforts of several standards groups and proven application experience to encourage significant product offerings.

REFERENCES

[1] J. C. Eidson, M. C. Fischer, and J. White, 2003, *"IEEE-1588 Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems"* in Proceedings of the 34ᵗʰ Annual Precise Time and Time Interval (PTTI) Systems and Applications Meeting, 3-5 December 2002, Reston, Virginia, USA (U.S. Naval Observatory, Washington, D.C.), pp. 243-254.

[2] M. E. Shepard, D. G. Fowley, R. L. Jackson, and D. B. King, 2003, *"Implementation of IEEE Sdc.-1588 in a Networked I/O Node,"* in NIST publication NISTIR 7070, Workshop on IEEE-1588, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, 24 September 2003, Gaithersburg, Maryland.

[3] D. S. Mohl, 2003, *"IEEE 1588 Network Devices,"* in NIST publication NISTIR 7070, Workshop on IEEE-1588, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, 24 September 2003, Gaithersburg, Maryland.

[4] D. Vook, B. Hamilton, A. Fernandez, J. Burch, and V. Srikantam, 2005, "*Update on High Precision Time Synchronization*" in the Proceedings of the 2005 Conference on IEEE-1588, 10-12 October 2005, Zurich, Switzerland.

[5] M. J. Teener, 2006, "*Ethernet Bridging and Ethernet AV™*", *http://www3.ietf.org/proceedings/06nov/ slides/tsvarea-1.pdf*

[6] J. C. Eidson and D. Pleasant, 2006, "*Timescales and IEEE 1588- Part 2,*" **LXI Connexion Magazine**, October 2006.

[7] J. Dai, 2003, "*Time correlation using network based data acquisition on-board a military test vehicle,*" in NIST publication NISTIR 7070, Workshop on IEEE-1588, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, 24 September 2003, Gaithersburg, Maryland.

[8] J. MacKay, 2004, "*Interfacing Mil Standard Equipment to an IEEE 1588 Enabled Ethernet Network,*" in NIST publication NISTIR 7192, 2004 Conference on IEEE-1588, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, 27-29 September 2004, Gaithersburg, Maryland.

[9] K. O'Donoghue, *et al.*, 2006, "*IEEE 1588 in Navy Next Generation Time Sensitive Applications*", in Proceedings of the 2006 Conference on IEEE-1588, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, 2-4 October 2006, Gaithersburg, Maryland.

[10] IEEE 1646-2004, "*IEEE Standard Communication Delivery Time Performance Requirements for Electric Power Substation Automation.*"

[11] G. Algie, 2004, "*Telecom Update to IEEE 1588 Workshop*", in NIST publication NISTIR 7192, 2004 Conference on IEEE-1588, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, 27-29 September 2004, Gaithersburg, Maryland.

[12] D. Tonks, 2005, "*IEEE 1588 in Telecommunications and Field Trial Results*", in Proceedings of the 2005 Conference on IEEE-1588, 10-12 October 2005, Zurich, Switzerland.

[13] S. Rodrigues, 2006, "*Combining Synchronous Ethernet and IEEE 1588 for use in Telecom,*" 2006 Conference on IEEE-1588, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, 2-4 October 2006, Gaithersburg, Maryland.

[14] C. Curry, 2006, "*Global Telecom Network Timing Standards- Decision Day Draws Close!*" WSTS Review 2006, *http://www.chronos.co.uk/pdfs/tel/WSTS_Review_2006.pdf*.

[15] M. Christer, S. Passe, and S. Stein, 2006, "*Design of an IEEE-1588 Interface for Sub-nanosecond Performance,*" in Proceedings of the 2006 Conference on IEEE-1588, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, 2-4 October 2006, Gaithersburg, Maryland.
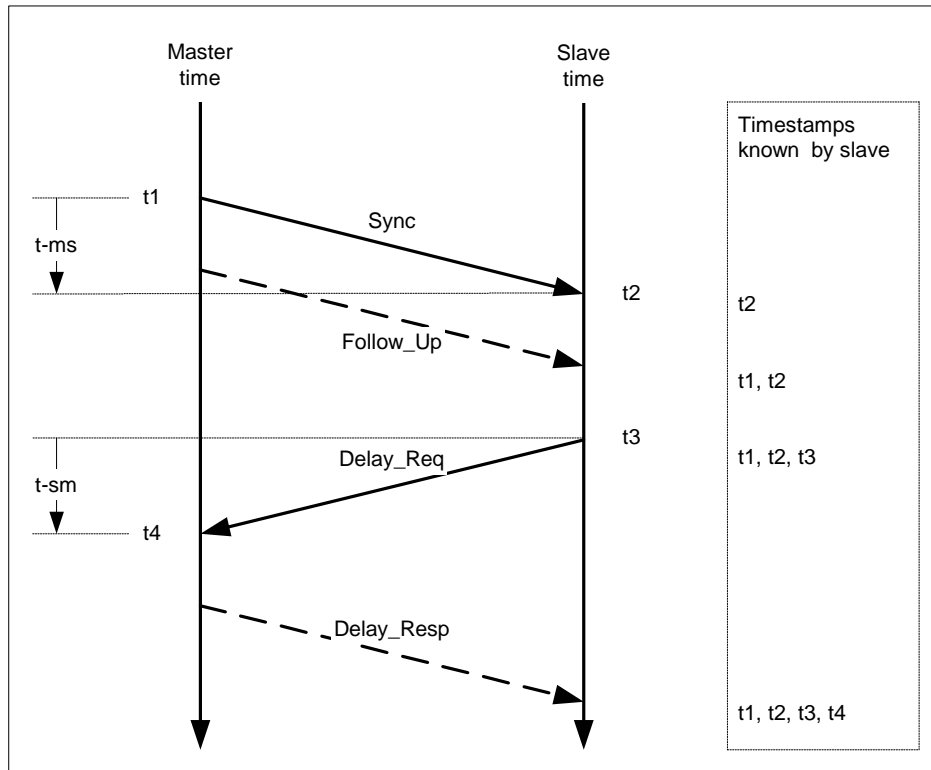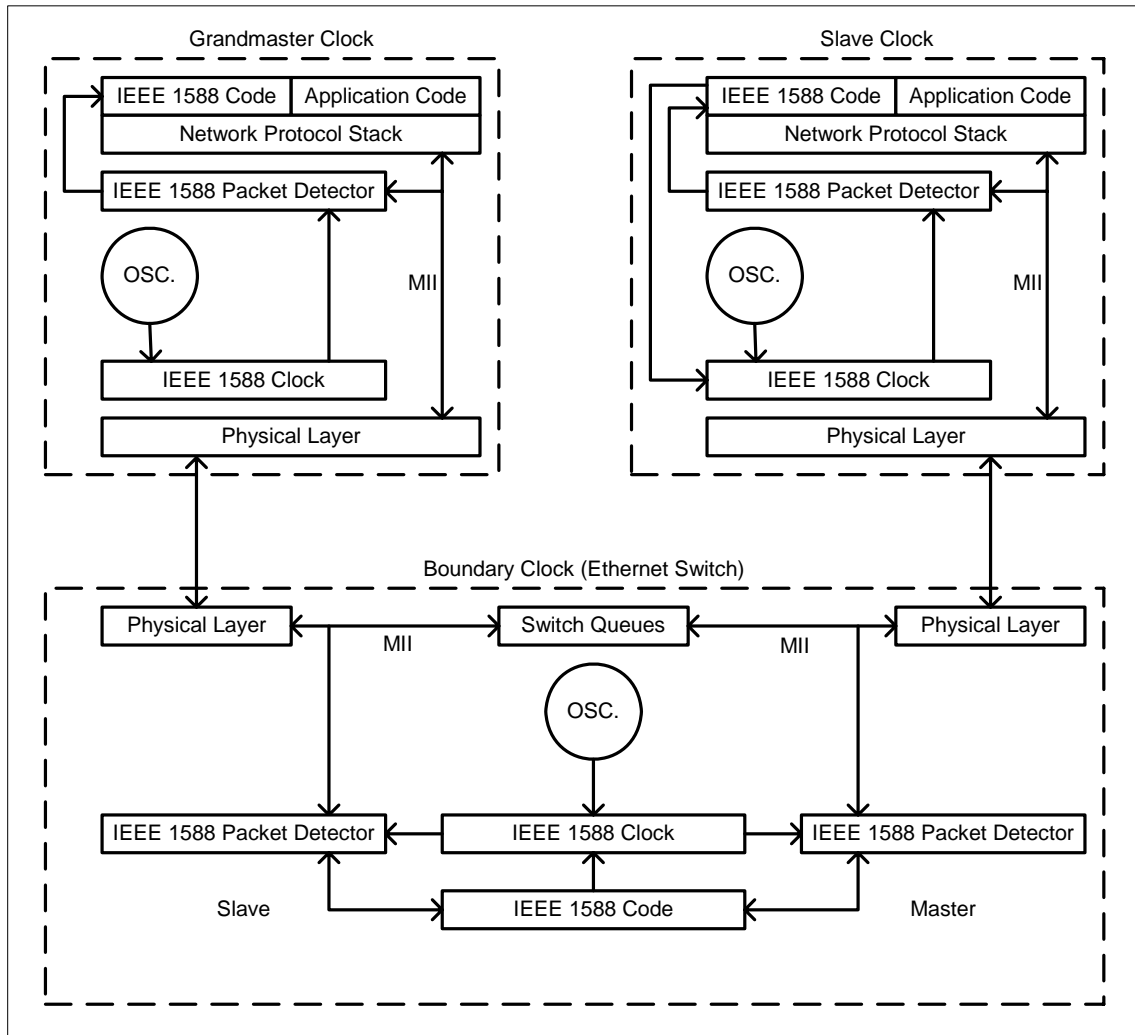
Figure 1. Synchronization timing diagram.

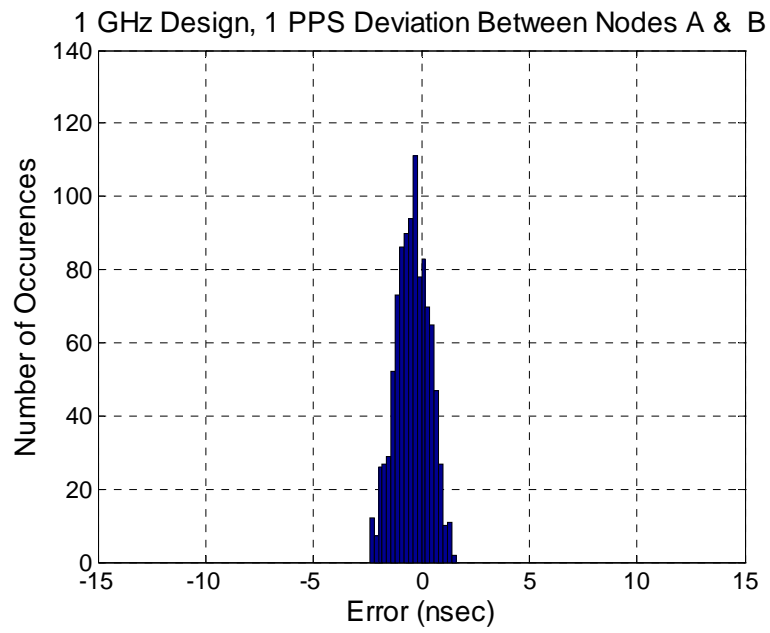Figure 2. Block diagram of 1588 devices.

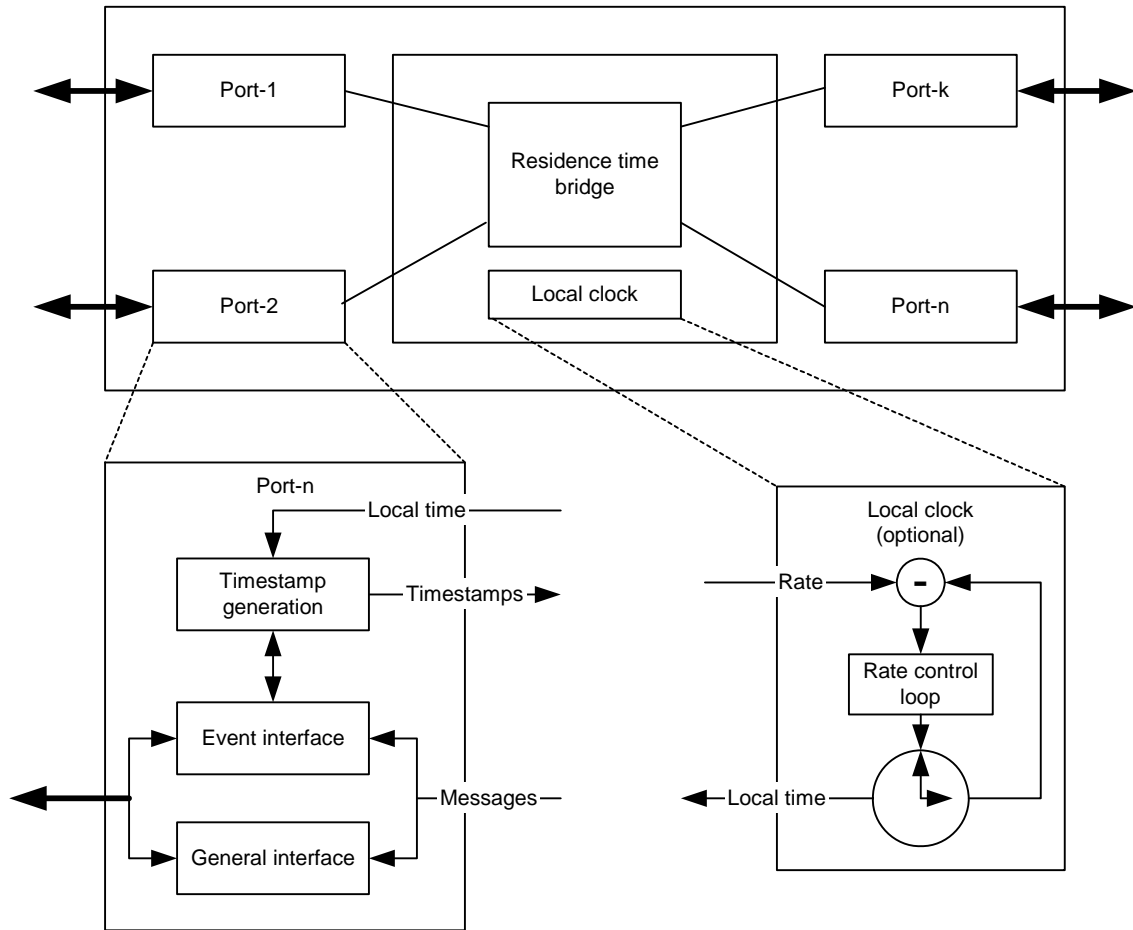Figure 3. Synchronization between two high performance clocks.
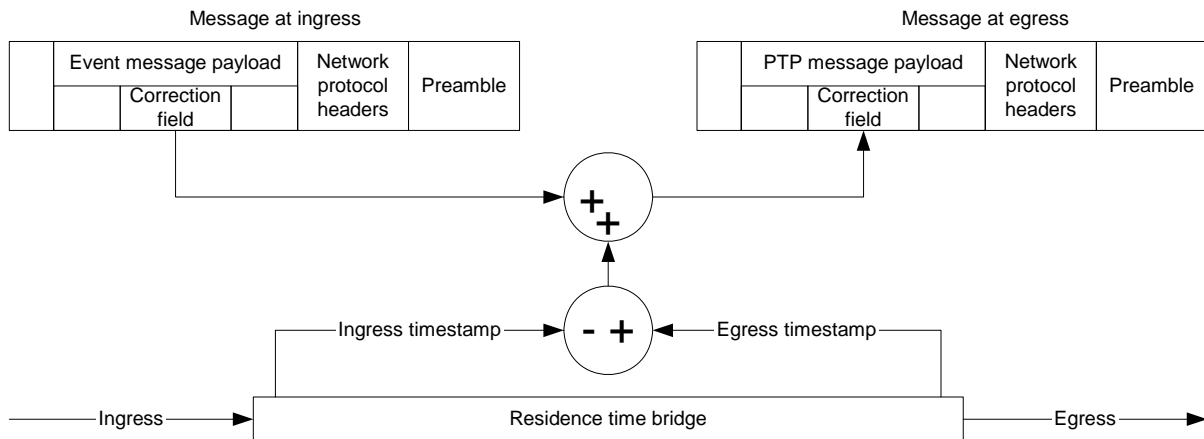
Figure 4. End-to-end transparent clock.

Figure 5. Residence time correction in an end-to-end transparent clock.
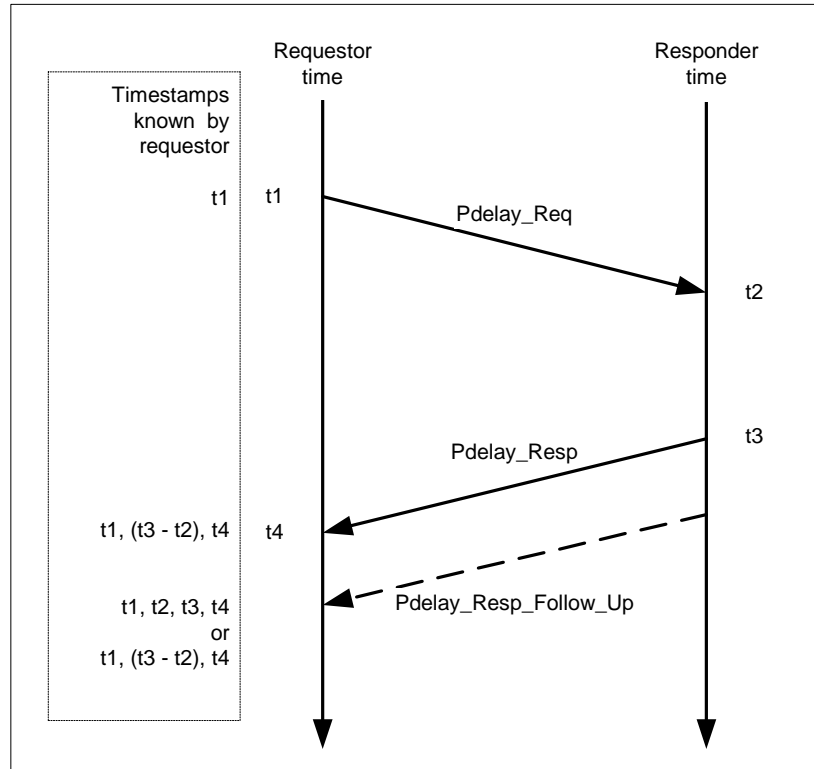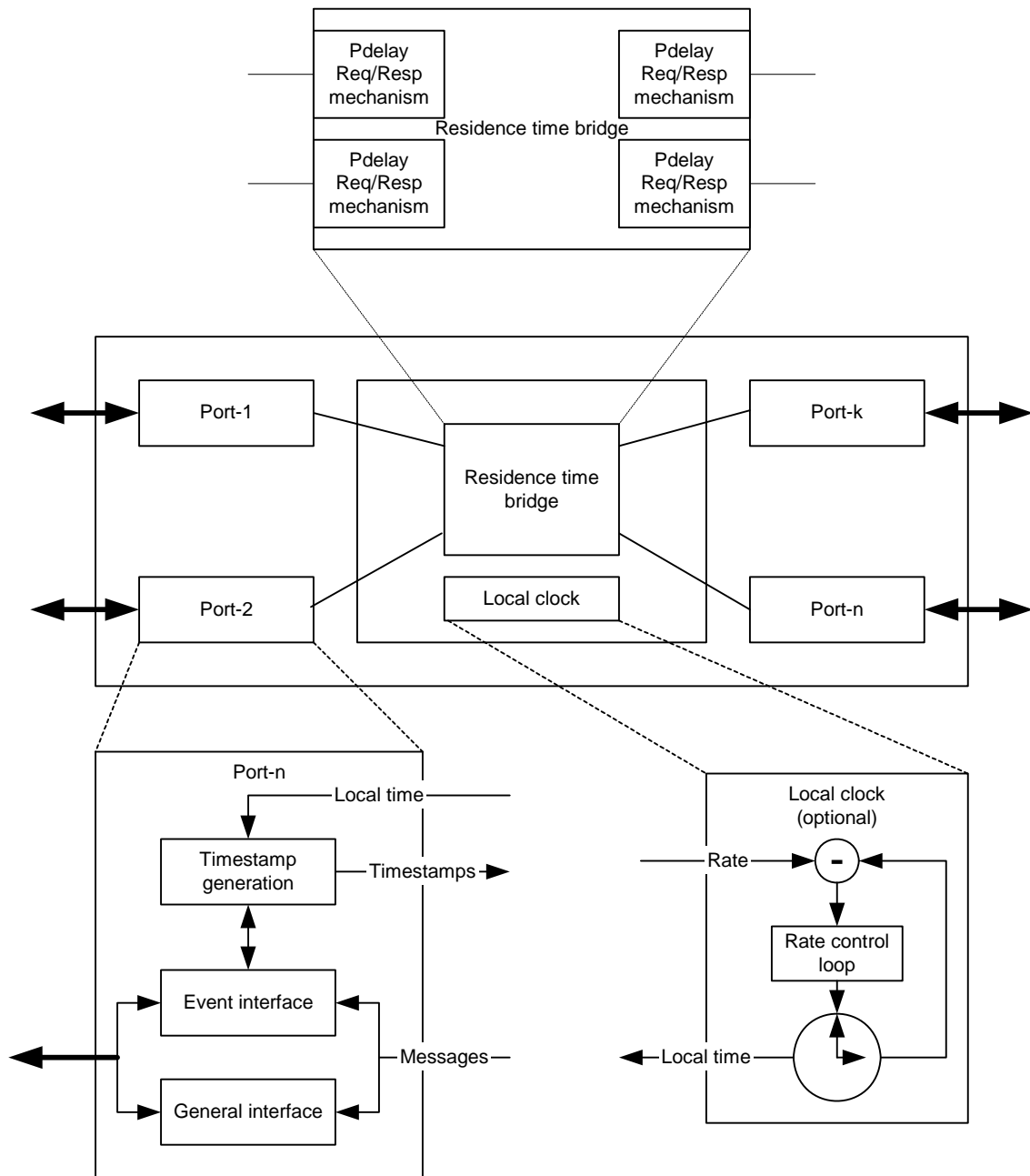
Figure 6. Peer-to-peer message timing diagram.

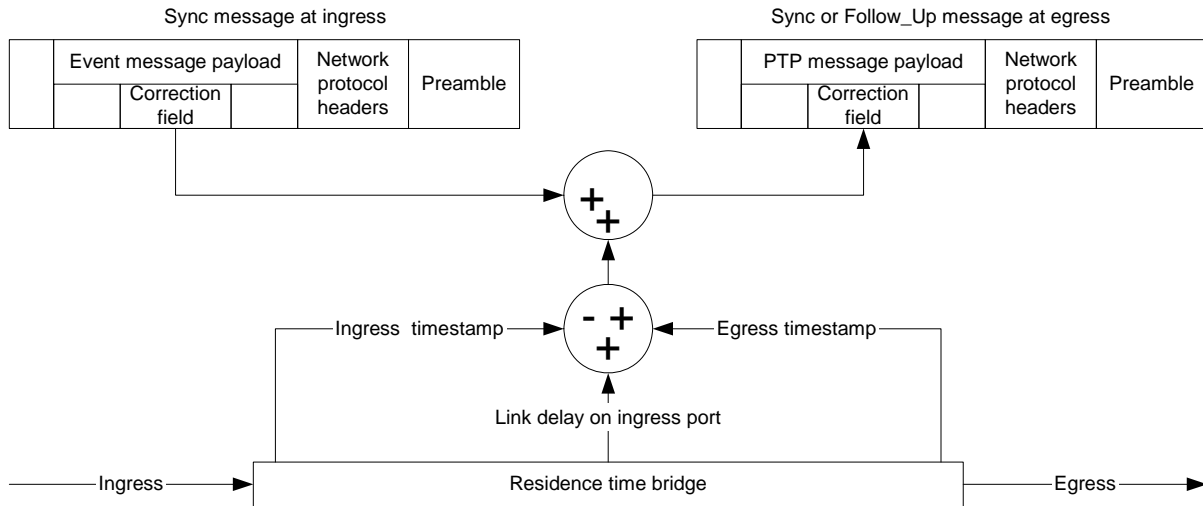Figure 7. Peer-to-peer transparent clock.

Figure 8. Residence time correction in a peer-to-peer transparent clock.
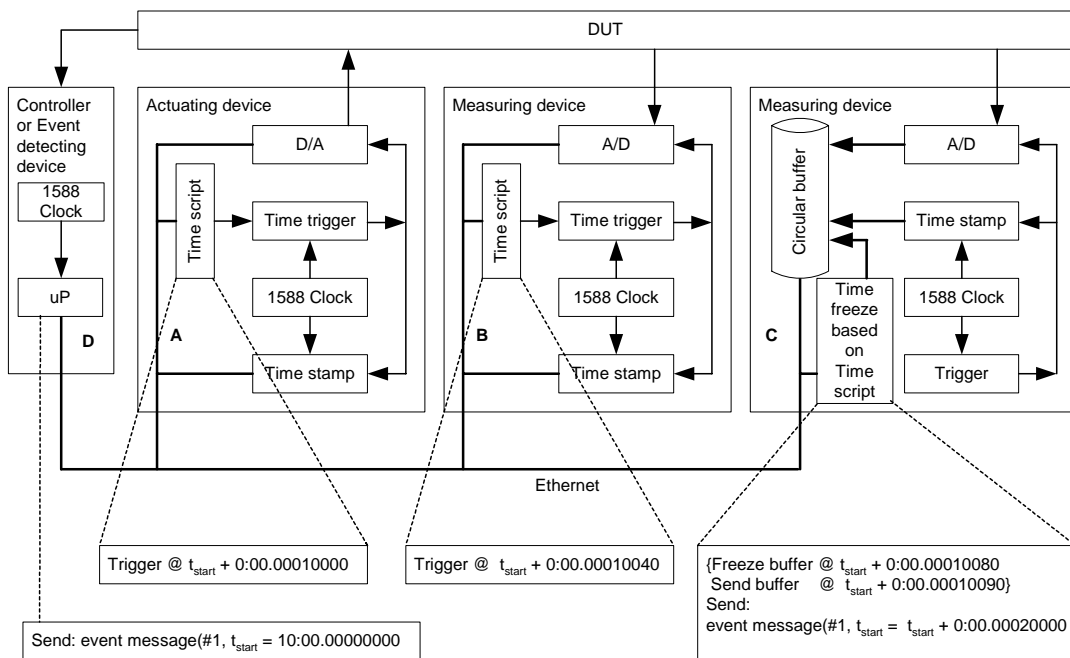
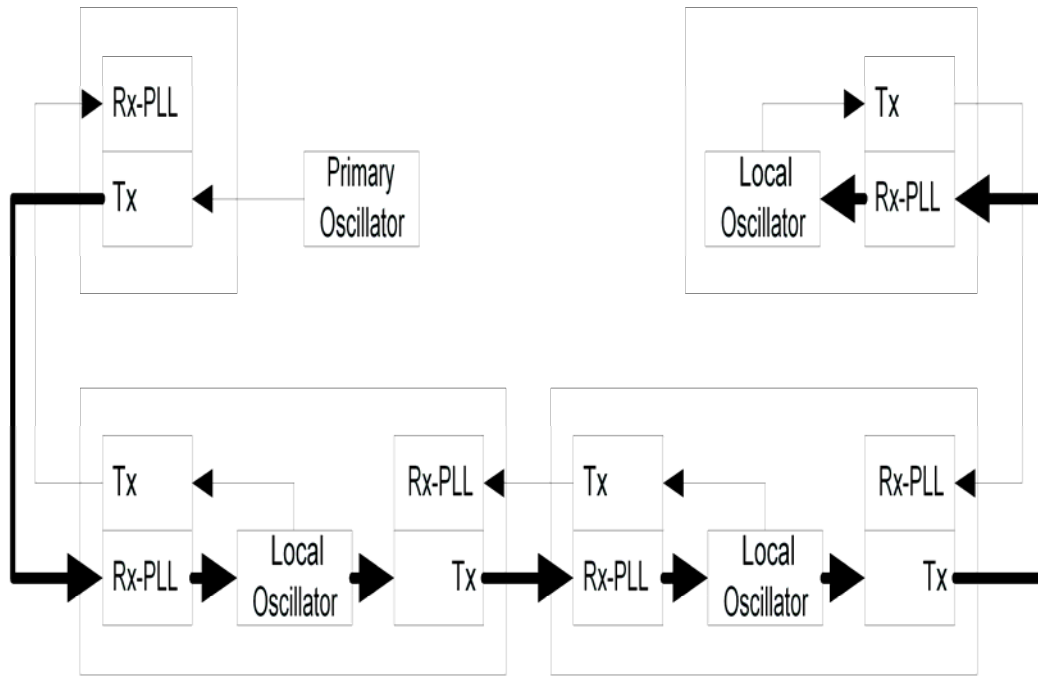Figure 9. Scheduled events in a Class B LXI test and measurement system.

Figure 10. Frequency transfer at the Ethernet physical layer.